

МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Автоматизированные системы управления»

СКРИПТОВЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ

*Методические рекомендации к лабораторным работам
для студентов специальности 1-40 05 01-01 «Информационные системы и
технологии (в проектировании и производстве)» очной и заочной форм
обучения*



Могилев 2022

УДК 519.682.2

ББК 32.973.4

Д48

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Автоматизированные системы управления»
«__» ноября 2022 г., протокол № 5

Составитель: ст. преподаватель Н. В. Выговская

Рецензент ???

Методические рекомендации предназначены к лабораторным занятиям студентов направления подготовки 1-40 05 01-01 «Информационные системы и технологии (в проектировании и производстве)» очной и заочной форм обучения.

Учебно-методическое издание

СКРИПТОВЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ

Ответственный за выпуск А. И. Якимов

Корректор

Компьютерная верстка

Подписано в печать . Формат 60 × 84/16. Бумага офсетная. Гарнитура Таймс.

Печать трафаретная. Усл. печ. л. . Уч.-изд. л. . Тираж 21 экз. Заказ №

Издатель и полиграфическое исполнение:
Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет».

Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/156 от 07.03.2019.

Пр-т Мира, 43, 212022, г. Могилев.

© Белорусско-Российский
университет, 2022

Содержание

Введение.....	3
2 Лабораторная работа № 2. Создание простых скриптов на JavaScript.....	4
3 Лабораторная работа № 3. Функции и обработка события на JavaScript...	5
4 Лабораторная работа № 4. Операторы ветвлений и циклов, логические операции в JavaScript	7
5 Лабораторная работа № 5. Методы в JavaScript	10
6 Лабораторная работа № 6. Обработка массивов на JavaScript.....	15
7 Лабораторная работа № 7. Использование элементов управления на JavaScript	16

Введение

При изучении дисциплины «Скриптовые языки программирования» студенты выполняют лабораторные работы, варианты которых приведены в данных методических рекомендациях.

Каждая лабораторная работа соответствует темам лекций и содержит в себе практические задания.

К защите работы студент подготавливает отчет, включающий в себя титульный лист, формулировку задания, описание исходных, результирующих данных, алгоритм решения задачи, текст программы и результаты ее тестирования.

Защита лабораторной работы состоит из двух частей: практической и теоретической. В практической части студент объясняет принципы работы представленной им программы, в теоретической – отвечает на вопросы по теме лабораторной работы.

При подготовке к защите студенту рекомендуется ответить на контрольные вопросы.

2 Лабораторная работа № 2. Создание простых скриптов на JavaScript

Цель работы: изучить основы разработки, подключение скриптов (сценариев) на языке JavaScript.

Порядок выполнения работы

- 1 Изучить теоретические сведения.
- 2 Выполнить задания, приуроченные к лабораторной работе.
- 3 Оформить отчет.

Требования к отчету

- 1 Цель работы.
- 2 Постановка задачи.
- 3 Результаты выполнения, тестирования и разработки программы.

Задания

- 1 Составить сценарий, в котором вычисляется площадь круга по заданному радиусу.
- 2 Составить сценарий, вычисляющий гипотенузу по заданным катетам.
- 3 Изменить любой скрипт добавив в него диалоговые окна для ввода исходных данных (функция `prompt`). Использовать функцию `alert` для вывода результата.

Пример выполнения задания

Пример 1. Вычисление площади треугольника

```
<script>
  /*Инициализируются две переменные*/
  const a = 8;
  const h = 10;
  /*Для формирования вывода используется метод write объекта
  document*/
  document.write("Площадь прямоугольного треугольника равна ", a * h / 2,
  ".");
</script>
```

Контрольные вопросы

- 1 Что представляет из себя сценарий языка JavaScript?
- 2 Что такое литералы?
- 3 Дайте определение термину «переменная». Приведите пример объявления переменной в JavaScript.
- 4 Как задать однострочный и многострочный комментарии в программе?

5 Как подключить сценарий языка JavaScript в HTML-документ?

3 Лабораторная работа № 3. Функции и обработка события на JavaScript

Цель работы: изучить механизм работы функций и обработки событий на языке JavaScript.

Порядок выполнения работы

- 1 Изучить теоретические сведения.
- 2 Выполнить задания, приуроченные к лабораторной работе.
- 3 Оформить отчет.

Требования к отчету

- 1 Цель работы.
- 2 Постановка задачи.
- 3 Результаты выполнения, тестирования и разработки программы.

Задания

- 1 На плоскости заданы координаты трех точек. Напишите сценарий, который вычисляет площадь треугольника (использовать событие Focus).
- 2 Напишите сценарий, который для точки, заданной координатами на плоскости, определяет расстояние до начала координат (использовать событие Select).
- 3 Напишите сценарий, который обменивает местами значения двух введенных переменных (использовать событие Blur).

Пример выполнения задания

Пример 1. Вычисление площади квадрата

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Обработка события Change - изменение значения элемента</title>
  <style>
    * {
      font-family: Verdana, Geneva, Tahoma, sans-serif;
      font-size: 16px;
    }

    p {
      color: rgb(104, 32, 172);
      font-size: 18px;
```

```

    }

    #reset-button {
        margin: 10px 0px;
        padding: 8px;
        border: none;
        border-radius: 5px;
        background-color: rgb(228, 199, 255);
    }

    #reset-button:hover {
        cursor: pointer;
        background-color: rgb(104, 32, 172);
        color: white;
    }
</style>
</head>

<body>
    <p>Вычисление площади квадрата</p>
    <form>
        Сторона: <input type="text" size="7" id="side"
onChange="calcSquareArea()" /> <br /><br />
        Площадь: <input type="text" size="7" id="result" /> <br /><br />
        <input id="reset-button" type="reset" value="Обновить" />
    </form>

    <script>
        const sideInput = document.querySelector('#side');
        const resultField = document.querySelector('#result');

        function calcSquareArea() {
            const sideLength = sideInput.value;
            resultField.value = sideLength ** 2;
        }
    </script>
</body>

</html>

```

Контрольные вопросы

- 1 Сколько способов объявления функций существует в JavaScript?
- 2 Приведите пример вызова функции. Что такое параметры функции?
- 3 Для чего используются функции в языке JavaScript?

4 Что из себя представляют DOM-события? Назовите часто используемые DOM-события.

4 Лабораторная работа № 4. Операторы ветвлений и циклов, логические операции в JavaScript

Цель работы: изучить операторы ветвлений, циклов и логические операции в сценариях языка JavaScript.

Порядок выполнения работы

- 1 Изучить теоретические сведения.
- 2 Выполнить задания, приуроченные к лабораторной работе.
- 3 Оформить отчет.

Требования к отчету

- 1 Цель работы.
- 2 Постановка задачи.
- 3 Результаты выполнения, тестирования и разработки программы.

Задания

- 1 Напишите программу, которая «переворачивает» заданное натуральное число.
- 2 Напишите сценарий, в котором определяется количество «счастливых» шестизначных автобусных билетов, т. е. таких, в номерах которых сумма первых трех цифр равна сумме трех последних.
- 3 Напишите программу, определяющую все делители заданного натурального числа.

Пример выполнения задания

Пример 1. Нахождение максимального значения введенных чисел

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <style>
      * {
        font-family: 'Comic Sans MS', 'Comic Sans', cursive;
      }

      form * {
```



```

    font-size: 18px;
}

h2 {
    text-align: center;
}

form {
    background: linear-gradient(45deg, rgba(211, 255, 167, 0.5), rgba(255,
236, 125, 0.5), rgba(255, 255, 255, 0.5), rgba(255, 236, 125, 0.5), rgba(211, 255,
167, 0.5));
    width: 50%;
    margin: 20px auto;
    padding: 20px;
    border-radius: 18px;
}

input {
    margin: 8px 5px;
}

.button {
    border: 1px solid rgb(151, 151, 151);
    border-radius: 5px;
    padding: 8px;
    background-color: rgba(255, 255, 255, 0.5);
}

.button:hover {
    background-color: white;
    cursor: pointer;
}
</style>
</head>

<body>
<h2>Вычисление максимального значения</h2>
<form>
    <label>Число 1:</label>
    <input type="number" id="num1" />
    <br />
    <label>Число 2:</label>
    <input type="number" id="num2" />
    <br />
    <label>Число 3:</label>

```

```

<input type="number" id="num3" />
<br />
<br />
<label>Максимальное значение равно </label>
<br />
<input type="number" size="14" id="result" />
<input class="button" type="button" value="Определить"
onClick="getMaxValue()" />
<br />
<input class="button" type="reset" />
</form>

<script>
const num1 = document.querySelector('#num1');
const num2 = document.querySelector('#num2');
const num3 = document.querySelector('#num3');
const resultField = document.querySelector('#result');

function getMaxValue() {
  const a = Number(num1.value);
  const b = Number(num2.value);
  const c = Number(num3.value);
  let m = a;
  if (b > m) {
    m = b;
  }
  if (c > m) {
    m = c;
  }

  resultField.value = m;
}
</script>
</body>
</html>

```

Пример 2. Определение свойств элемента формы

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

```

```

<title>Операции над объектами. Свойства текстового поля</title>
</head>

<body>
  <h4>
    Определение свойств объектов
  </h4>
  <form id="form1">
    <input type="button" value=Выполнить id="run-button">
    <hr>
    <textarea name="data" cols=30 rows=10 id=1>
      Текст
    </textarea>
    <hr>
    <input type="reset" value=Очистить>
    <script>
      const form = document.querySelector('#form1');
      const runButton = document.querySelector('#run-button');
      runButton.addEventListener('click', () => propobj(form));

      function propobj(obj) {
        var result = ""
        for (var i in obj) { result += obj.data.value + "." + i + " = " + (obj.data)[i] +
"\r\n" }
        result += "\n\r"
        form1.data.value = result
      }
    </script>

  </form>
</body>
</html>

```

Контрольные вопросы

- 1 Какой оператор используется для организации ветвлений в JavaScript?
- 2 Объясните механизм выполнения условного оператора.
- 3 Для чего используется конструкция else if (условие)?
- 4 Что такое вложенные циклы? Приведите пример вложенного цикла.
- 5 Для чего используются операторы continue и break?
- 6 Объясните механизм работы конструкции while и сравните его с работой цикла do...while.

5 Лабораторная работа № 5. Методы в JavaScript

Цель работы: изучить структуру модели HTML DOM (Document Object Model), научиться работать с методами объектов на языке JavaScript.

Порядок выполнения работы

- 1 Изучить теоретические сведения.
- 2 Выполнить задания, приуроченные к лабораторной работе.
- 3 Оформить отчет.

Требования к отчету

- 1 Цель работы.
- 2 Постановка задачи.
- 3 Результаты выполнения, тестирования и разработки программы.

Задания

- 1 Написать сценарий выбора из трех изображений одного, которое вставляется ниже этих трех.
- 2 Написать сценарий картинки с "эффектом приближения", т.е. увеличения размеров как реакция на попадание курсора мыши в поле рисунка (использовать свойства width и height).
- 3 Написать сценарий графического горизонтального меню с появляющейся стрелкой над пунктом, у которого находится курсор.

Пример выполнения задания

Пример 1. Перестановка изображений местами

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Document</title>
  <style>
    * {
      font-family: Verdana, Geneva, Tahoma, sans-serif;
      font-size: 18px;
    }

    input[type="button"] {
      background-color: antiquewhite;
      border: 2px solid burlywood;
    }

    img {
      width: 100px;
      margin: 5px;
```

```

    border: 1px solid black;
    background-color: beige;
  }
</style>
</head>
<body>
  <h4>Галерея рисунков</h4>
  
  
  

  <form>
    <label>Рисунки с номерами:</label>
    <input type="text" id="first-image-number" size="1" />
    <input type="text" id="second-image-number" size="1" />
    <input type="button" value="Поменять местами" onClick="swapImages()"
  />
  </form>

  <script>
    const firstImageNumberInput = document.querySelector('#first-image-
number');
    const secondImageNumberInput = document.querySelector('#second-image-
number');

    function swapImages() {
      const firstImageNumber = +firstImageNumberInput.value;
      const secondImageNumber = +secondImageNumberInput.value;
      const images = document.images;

      if (firstImageNumber < 1 || firstImageNumber > 3 || secondImageNumber <
1 || secondImageNumber > 3) {
        alert('Неверно заданы номера рисунков!');
      }

      const firstImage = images[firstImageNumber - 1];
      const secondImage = images[secondImageNumber - 1];
      [firstImage.src, secondImage.src] = [secondImage.src, firstImage.src];
    }
  </script>
</body>
</html>

```

Пример 2. Простое вертикальное меню

Содержимое файла *script.js*:

```
const navItems = document.querySelectorAll("nav>ul>li");

navItems.forEach((item) => {
  item.onmouseover = itemMouseoverHandler;
  item.onmouseout = itemMouseoutHandler;
});

function itemMouseoverHandler() {
  this.firstElementChild.classList.remove('hide');
}

function itemMouseoutHandler() {
  this.firstElementChild.classList.add('hide');
}
```

Содержимое файла *index.html*:

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Document</title>
  <link rel="stylesheet" href="./style.css" />
</head>

<body>
  <nav>
    <ul>
      <li>
        
        <a href="#">Home</a>
      </li>
      <li>
        
        <a href="#">About</a>
      </li>
      <li>
        
        <a href="#">Clients</a>
      </li>
    </ul>
  </nav>
</body>
</html>
```

```

    <li>
      
      <a href="#">Contact Us</a>
    </li>
  </ul>
</nav>
<script src="./script.js"></script>
</body>

</html>

```

Содержимое файл *style.css*:

```

* {
  padding: 0;
  margin: 0;
  font-family: "Comic Sans MS", "Comic Sans", cursive;
}

```

```

body{
  display: flex;
  height: 100vh;
}

```

```

nav {
  background: white;
  box-shadow: 0 3px 10px -2px rgba(0, 0, 0, 0.1);
  border: 1px solid rgba(0, 0, 0, 0.3);
  margin: 50px;
  padding: 5px;
  padding-right: 30px;
}

```

```

nav ul {
  width: 140px;
  height: 100%;
  display: flex;
  justify-content: space-evenly;
  flex-direction: column;
  list-style: none;
}

```

```

nav ul li {
  display: flex;
  justify-content: center;
}

```

```

    align-items: center;
}

nav ul li a {
    display: block;
    color: black;
    font-size: 0.9em;
    text-decoration: none;
    text-transform: uppercase;
}

img{
    rotate: 90deg;
    max-height: 25px;
    max-width: 25px;
}

.hide {
    visibility: hidden;
}

```

Контрольные вопросы

- 1 Дайте определение термину «объект». Что такое свойства и методы объекта?
- 2 Что такое объектная модель документа (Document Object Model)? Назовите некоторые методы и свойства HTML DOM.
- 3 Приведите пример осуществления доступа к методу объекта.
- 4 Объясните механизм поведения ключевого слова this в JavaScript.

6 Лабораторная работа № 6. Обработка массивов на JavaScript

Цель работы: изучить способы объявления, инициализации и работы с массивами на языке JavaScript.

Порядок выполнения работы

- 1 Изучить теоретические сведения.
- 2 Выполнить задание, приуроченное к лабораторной работе.
- 3 Оформить отчет.

Требования к отчету

- 1 Цель работы.
- 2 Постановка задачи.
- 3 Результаты выполнения, тестирования и разработки программы.

Задания

- 1 Создать простейший мультипликационный сюжет с использованием массива.
- 2 Задан одномерный массив вещественных чисел. Напишите сценарий, который определяет число положительных элементов массива.
- 3 Задан одномерный массив вещественных чисел. Напишите сценарий, позволяющий найти максимальный элемент в массиве.

Контрольные вопросы

- 1 Дайте определение термину «массив». Для чего используются массивы в JavaScript?
- 2 Приведите примеры создания массивов в JavaScript.
- 3 Что такое одномерные и многомерные массивы?
- 4 Перечислите основные методы работы с массивами в JavaScript.
- 5 Какое свойство используется для получения длины массива?

7 Лабораторная работа № 7. Использование элементов управления на JavaScript

Цель работы: изучить стандартные элементы управления («переключатель», «флажок») и научиться работать с ними на языке JavaScript.

Порядок выполнения работы

- 1 Изучить теоретические сведения.
- 2 Выполнить задание, приуроченное к лабораторной работе.
- 3 Оформить отчет.

Требования к отчету

- 1 Цель работы.
- 2 Постановка задачи.
- 3 Результаты выполнения, тестирования и разработки программы.

Задание

Напишите сценарий обработки анкеты слушателя курсов. Пользователь может выбрать курс, его продолжительность, язык, на котором он готов работать с преподавателем, и форму отчетности. В зависимости от этих параметров определяется стоимость отдельного курса и стоимость всего обучения.

ФИО *данное поле обязательно для заполнения

e-mail

Возраст

Адрес

Название курсов	Продолжительность	Язык	Отчетность	Стоимость
<input checked="" type="checkbox"/> Информатика	<input type="radio"/> 36 <input type="radio"/> 64 <input type="radio"/> 128	<input type="radio"/> Русский <input type="radio"/> Английский	<input type="radio"/> Экзамен <input type="radio"/> Зачет	Стоимость 18 у.е.
<input checked="" type="checkbox"/> Базы данных	<input type="radio"/> 36 <input type="radio"/> 64 <input type="radio"/> 128	<input type="radio"/> Русский <input type="radio"/> Английский	<input type="radio"/> Экзамен <input type="radio"/> Зачет	Стоимость 6 у.е.
<input type="checkbox"/> Сети	<input type="radio"/> 36 <input type="radio"/> 64 <input type="radio"/> 128	<input type="radio"/> Русский <input type="radio"/> Английский	<input type="radio"/> Экзамен <input type="radio"/> Зачет	Стоимость 0 у.е.
<input type="checkbox"/> Логика	<input type="radio"/> 36 <input type="radio"/> 64 <input type="radio"/> 128	<input type="radio"/> Русский <input type="radio"/> Английский	<input type="radio"/> Экзамен <input type="radio"/> Зачет	Стоимость 0 у.е.
Общая стоимость 24 у.е.				

Рисунок 7.1 – Сценарий анкеты слушателя курсов

Пример выполнения задания**Пример 1. Сценарий «Анкета переводчика»**

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Данные, представленные флажком. Анкета переводчика</title>
```

```
<style>
```

```
* {
```

```
font-family: Cambria, Cochin, Georgia, Times, 'Times New Roman', serif;
```

```
}
```

```
section {
```

```
display: flex;
```

```
flex-direction: column;
```

```
align-items: center;
```

```
}
```

```
form :first-child {
```

```
margin-bottom: 10px;
```

```
display: block;
```

```
}
```

```
form {
```

```
border: 4px dashed gainsboro;
padding: 20px;
font-size: 18px;
}
```

```
input[type="text"] {
  font-size: 18px;
  width: 100%;
  box-sizing: border-box;
}
```

```
input[type="checkbox"] {
  margin: 8px 0;
}
```

```
.button {
  font-size: 18px;
  margin: 20px 0;
  padding: 8px;
  border: none;
  border-radius: 5px;
  background-color: rgb(168, 179, 255);
  cursor: pointer;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<section>
```

```
<h3>Анкета для переводчиков</h3>
```

```
<form name="form1">
```

```
<label>
```

Укажите те языки, которыми Вы владеете в совершенстве:

```
</label>
```

```
<input type="checkbox" name="lan" value=100>
```

```
<label>русский</label>
```

```
<br>
```

```
<input type="checkbox" name="lan" value=200>
```

```
<label>английский</label>
```

```
<br>
```

```
<input type="checkbox" name="lan" value=300>
```

```
<label>французский</label>
```

```

<br>

        <input class="button" type="button" value=Вознаграждение
onClick="grant()">
    <br>

    <input type="text" class="result">
    <br>

    <input class="button" type="reset" value="Отменить">
</form>
</section>

<script language="JavaScript">

function grant() {
    const checkboxes = document.querySelectorAll('input[type="checkbox"]');

    let award = 0;
    checkboxes.forEach((checkbox) => {
        if (checkbox.checked) {
            award = award + Number(checkbox.value)
        }
    });

    document.querySelector('.result').value = "Вам полагается вознаграждение
" + award + " у.е."
}

</script>

</body>
</html>

```

Пример 2. Сценарий «Анкета читателя»

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Анкета читателя</title>
</head>

```

```

<style>
* {
  font-family: Cambria, Cochin, Georgia, Times, 'Times New Roman', serif;
}

h3 {
  text-align: center;
}

h4 {
  font-style: italic;
}

div {
  font-size: 17px;
  border-radius: 5px;
}

div input {
  margin: 8px 0;
}

div label {
  color: rgb(117, 0, 167);
  margin-left: 8px;
}

form {
  box-sizing: border-box;
  margin: 10px;
  padding: 0 20px 15px;
}

.second-form {
  border: 4px dashed gainsboro;
  padding: 0 20px 20px;
}

.first-form input {
  width: 100%;
  box-sizing: border-box;
}

textarea {
  margin: 30px 0 15px;
}

```

```

    font-size: 17px;
}

.button {
    font-size: 16px;
    padding: 10px 20px;
    border: none;
    border-radius: 5px;
    background-color: rgb(235, 189, 255);
    cursor: pointer;
}

section {
    display: flex;
    flex-direction: column;
    justify-content: center;
    align-items: center;
}
</style>

<body>
  <h3>Анкета читателя</h3>
  <section>
    <form class="first-form">
      <h4>
        Введите название любимого журнала или газеты
      </h4>
      <input type="text"><br>
    </form>

    <form class="second-form">
      <h4>Что Вас привлекает в издании?</h4>
      <div>
        <input type="checkbox" value="Стиль подачи материала">
        <label>
          Стиль подачи материала
        </label>
        <br>
        <input type="checkbox" value="Достоверность информации">
        <label>
          Достоверность информации
        </label>
        <br>
        <input type="checkbox" value="Дизайн и оформление">
        <label>

```

```

        Дизайн и оформление
    </label>
    <br>
</div>
<textarea cols=50 rows=7> </textarea>
<br>
<input class='button' type="reset" value="Отмена">
</form>
</section>

<script>
    const checkboxes = document.querySelectorAll('.second-form
input[type="checkbox"]');

    checkboxes.forEach((checkbox) => { checkbox.addEventListener('click', set)
});

    const firstLine = "Вас привлекает:\n";
    let str = firstLine;

    function set() {
        if (this.checked) {
            str = str + this.value + "\n";
            document.querySelector('.second-form textarea').value = str;
        }
    }
</script>
</body>

</html>

```

Контрольные вопросы

- 1 Что представляют собой флажки и переключатели?
- 2 Какое событие генерируется при нажатии на флажок или переключатель?
- 3 Какое свойство используется для проверки состояния флажка?
- 4 Для чего используется атрибут name при создании групп переключателей (флажков)?
- 5 Какое свойство используется для получения значения выбранного переключателя?